

BAB III

LANDASAN TEORI

3.1. Sarang Semut

Myrmecodia pendans (sarang semut) merupakan tanaman epifit yang kaya akan *phytochemical*. *Myrmecodia pendans* (*genus myrmecophytes*), juga dikenal penduduk asli Papua sebagai sarang semut Sarang Semut (*Myrmecodia pendans*) sejenis tumbuhan yang menempel pada tumbuhan lain yang lebih besar. Tumbuhan Sarang Semut umumnya banyak dijumpai di daerah Kalimantan, Sumatera, Papua Nugini, Filipina, Kamboja, Malaysia, Cape York, Kepulauan Solomon dan Papua (A.F.S.L & H.T.W., 2009).

Klasifikasi ilmiah dari tumbuhan sarang semut adalah sebagai berikut (Subroto & Hendro , 2008) :

Divisi	: <i>Tracheophyta</i>
Kelas	: <i>Magnoliopsida</i>
Sub Kelas	: <i>Lamiidae</i>
Ordo	: <i>Rubiales</i>
Famili	: <i>Rubiaceae</i>
Genus	: <i>Myrmecodia</i>

Sarang semut memiliki keunikan yang terletak pada interaksi dari semut yang menjadikan lorong-lorong umbi sebagai sarang didalamnya dan membuat koloni sehingga semut-semut sangat betah bersarang di dalam tanaman ini. Sehingga dengan jangka waktu yang lama terjadi reaksi kimiawi secara alami antara senyawa

yang dikeluarkan semut dengan zat yang terkandung di dalamnya. Sarang Semut tidak memiliki akar tetapi menempel pada batang pohon. Efek negatif sarang semut belum ditemukan tetapi kebalikannya dapat meningkatkan fungsi metabolisme tubuh dan kelancaran dari peredaran darah meningkat sehingga stamina tubuh juga meningkat (Hertiani, et al., 2010).



Gambar 3. 1 Tumbuhan Sarang Semut yang menggantung pada pohon

Kandungan senyawa-senyawa kimia dari golongan flavonoid dan tannin yang dimiliki Sarang Semut diketahui mampu menyembuhkan berbagai macam penyakit. Flavonoid berfungsi sebagai antibiotik, antivirus untuk HIV dan herpes (Soeksmanto, et al., 2010). Selain itu juga *flavonoid* dimanfaatkan dalam mengobati dan mencegah beberapa penyakit seperti asma, katarak, diabetes, encok/rematik, migrain, wasir, periodontitis dan kanker. Sarang Semut diketahui juga mengandung senyawa antioksidan, vitamin, mineral dan asam formiat. Antioksidan pada semut

berperan dalam pembentukan koloni dan menjaga tempat telur jauh dari kuman penyakit.

Umumnya, bagian tumbuhan Sarang Semut yang digunakan sebagai obat adalah bagian *hypocotyl (caudex)* (Soeksmanto, et al., 2010). Dengan cara merebus bagian *hypocotyl* sarang semut yang sudah dikeringkan (Hertiani, et al., 2010) dapat digunakan sebagai obat. Melalui uji penapisan kimia, diperoleh sarang semut mengandung senyawa aktif golongan *flavonoid*.

3.2. Citra Digital

Citra merupakan representasi (gambaran), kemiripan, atau imitasi dari suatu objek. Citra terbagi menjadi dua, yaitu citra yang bersifat analog dan citra yang bersifat digital, citra analog adalah citra yang bersifat *continue* seperti gambar pada monitor televisi, data sinar X, dan lain-lain. Sedangkan pada citra digital adalah citra yang dapat diolah oleh komputer (Sutojo & Siswanto, 2014).

Citra dapat didefinisikan sebagai fungsi $f(x,y)$ yang berukuran M adalah baris dan N adalah kolom, dengan x dan y adalah koordinat spasial, dan amplitudo f di titik koordinat (x,y) dinamakan intensitas atau tingkat keabuan dari pada citra tersebut (Darma, 2010).

Citra digital merupakan representatif dari citra yang diambil oleh mesin dengan bentuk pendekatan berdasarkan sampling dan kuantisasi. Sampling menyatakan besarnya kotak-kotak yang disusun dalam baris dan kolom dengan kata lain, sampling pada citra menyatakan besar kecilnya ukuran pixel (titik) pada citra,

dan kuantisasi menyatakan besarnya nilai tingkat kecerahan yang dinyatakan dalam tingkat keabuan (grayscale) sesuai dengan jumlah bit biner yang digunakan oleh mesin, dengan kata lain kuantisasi pada citra menyatakan jumlah warna yang ada pada citra.

Ada beberapa cara menyimpan citra digital yang sering digunakan adalah:

1. Citra Biner (monokrom). Banyaknya dua warna, yaitu hitam dan putih. Dibutuhkan 1 bit di memori untuk menyimpan kedua warna ini.
2. Citra Grayscale (skala keabuan). Banyaknya warna tergantung pada jumlah bit yang disediakan di memori untuk menampung kebutuhan warna ini, citra 2 bit memiliki 4 warna, citra 3 bit memiliki 8 warna, dan seterusnya. Semakin besar jumlah bit warna yang disediakan oleh memori, semakin luas gradasi warna yang terbentuk.
3. Citra Warna (true color). Setiap piksel pada warna mewakili warna yang merupakan kombinasi dari tiga warna dasar (RGB = *Red Green Blue*). Setiap warna dasar menggunakan penyimpanan 8 bit = 1 byte, yang berarti setiap warna mempunyai gradasi sebanyak 255 warna. Berarti setiap piksel mempunyai kombinasi warna sebanyak $28 \times 28 \times 28 \times 224 = 16$ juta warna lebih. Itulah sebabnya dinamakan true color karena jumlah warna yang cukup besar sehingga bisa dikatakan hampir semua mencakup jumlah warna yang lebih besar sehingga bisa dikatakan hampir mencakup semua warna alam.

3.3. Resolusi dan Kuantisasi

Ukuran dari *grid pixel* 2D dan ukuran data dari masing-masing *pixel* citra menentukan resolusi spasial dan kuantisasi warna dari citra. Representasi dan ukuran dari suatu citra ditentukan oleh resolusinya. Dengan demikian resolusi dari sumber citra, dalam penelitian ini dimisalkan kamera dapat dibedakan dalam tiga kuantitas yaitu :

1. Resolusi spasial, merupakan jumlah *pixel* yang digunakan untuk menutupi jarak visual yang ditangkap pada citra. dimensi kolom (C) dikalikan baris (R), yang berhubungan dengan sampling sinyal citra dan resolusi digital dari citra. $C \times R$ seperti misalnya 640x480, 800x600, 1024x768.
2. Resolusi bit, adalah jumlah dari kemungkinan intensitas atau warna yang dimiliki oleh suatu *pixel* yang berhubungan dengan kuantisasi dari informasi citra.
3. Resolusi temporal, untuk melakukan sistem *capture* yang kontinu seperti video, merupakan angka dari citra yang ditangkap dalam suatu periode waktu tertentu.

3.4. Pengolahan Citra

Pengolahan citra (*image processing*) mempunyai tujuan sebagai berikut :

1. Memperbaiki kualitas citra, dimana citra yang dihasilkan dapat menampilkan informasi secara jelas ataupun dapat melihat informasi yang diharapkan dengan menginterpretasikan citra yang ada.

2. Mengekstraksi informasi ciri yang dominan pada suatu citra.

3.5. Pre-Processing

3.5.1. Proses Citra Warna

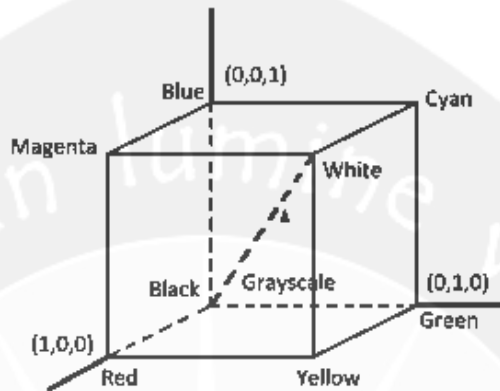
Manusia mampu melihat warna dikarenakan cahaya yang dipantulkan oleh suatu objek. Dimana spektrum cahaya kromatis berkisar antara 400-700 nm (Zhou, et al., 2010). Kromatis berarti kualitas warna cahaya ditentukan oleh panjang gelombang.

Ruang warna sebagai suatu spesifikasi sistem koordinat dan suatu subruang dalam sistem tersebut dengan setiap warna dinyatakan dengan satu titik didalamnya. Tujuan dibentuknya ruang warna adalah untuk memfasilitasi spesifikasi warna dalam bentuk suatu standar. Ruang warna paling dikenal pada perangkat komputer adalah RGB, yang sesuai dengan watak manusia dalam menangkap warna. Setelah adanya ruang warna RGB kemudian dibuat banyak ruang warna, antar lain HSI, CMY, LUV dan YIQ.

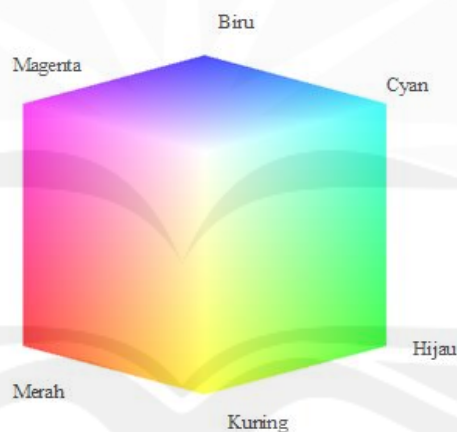
3.5.1.1. Ruang Warna RGB (Red, Green, Blue)

Ruang warna RGB bisa diterapkan pada monitor CRT dan kebanyakan sistem grafika komputer. Ruang warna ini menggunakan tiga komponen dasar yaitu merah (R), hijau (G), dan biru (B). Setiap piksel dibentuk oleh ketiga komponen tersebut. Model RGB biasa disajikan dalam bentuk kubus tiga dimensi, dengan warna merah, hijau dan biru berada pada pojok sumbu (Gambar 3.2). Warna hitam (black) berada pada titik asal dan warna putih (white) berada di ujung kubus yang berseberangan.

Gambar 3.3 memperlihatkan kubus warna secara nyata dengan resolusi 24 bit, dimana dengan menggunakan 24 bit jumlah warna mencapai 16.777.216.



Gambar 3. 2 Skema Warna RGB menurut Gonzales



Gambar 3. 3 Kubus Warna dengan 24 bit

Model warna RGB biasa digunakan karena untuk memperoleh kemudahan dalam perancangan *hardware*, tetapi sebenarnya tidak ideal untuk beberapa aplikasi. Hal ini dikarenakan mengingat warna merah, hijau, dan biru sesungguhnya terkolerasi erat, sangat sulit untuk beberapa algoritma pemrosesan citra (Crane, 1997).

3.5.1.2. Ruang Warna HSV

HSV (*Hue Saturation Value*) merupakan ruang warna yang merepresentasikan warna seperti yang dilihat oleh mata manusia. H berasal dari *Hue*, yaitu berupa sudut dari 0 sampai 360 derajat. S berasal dari *Saturasi*, merupakan ukuran seberapa besar kemurnian dari warna tersebut. Saturasi biasanya bernilai 0 sampai 1 (0% - 100%) dan menunjukkan nilai keabu-abuan warna dimana 0 menunjukkan abu-abu dan 1 menunjukkan warna primer murni. V berasal dari *Value* atau intensitas yaitu ukuran seberapa besar kecerahan suatu warna atau seberapa besar cahaya datang dari suatu warna, nilai value terdiri dari 0%-100%.

3.5.1.3. Citra Berskala Keabuan (Grayscale)

Citra berskala keabuan (grayscale) menangani gradasi warna hitam dan putih, yang menghasilkan efek warna abu-abu. Citra jenis berskala keabuan, warna dinyatakan dengan intensitas. Dalam hal ini, kisaran intensitas berkisar 0-255 dimana nilai 0 menyatakan hitam dan nilai 255 menyatakan putih.

3.5.1.4. Thresholding (Pengambangan)

Thresholding citra menjadi titik pusat dalam proses segmentasi sebuah citra. Thresholding akan menghasilkan citra biner, yaitu citra yang memiliki dua nilai tingkat keabuan berupa hitam dan putih. Bentuk umum proses thresholding adalah sebagai berikut :

$$g(x,y) = \begin{cases} 1, & \text{if } (x,y) \geq T \\ 0, & \text{if } (x,y) < T \end{cases}$$

Piksel yang diberi nilai 1 berkaitan dengan objek sedangkan piksel yang diberi nilai 0 berkaitan dengan background. Ketika T adalah konstanta, pendekatan ini dinamakan *global thresholding*.

3.5.2. Histogram

Histogram memegang peranan penting dalam proses pengolahan citra seperti *enhancement, compression, segmentation dan description*. Histogram sebuah citra digital dengan level intensitas pada range $[0, L-1]$ adalah fungsi diskrit :

$$h(r_k) = n_k$$

Dimana r_k adalah nilai intensitas k th dan n_k jumlah piksel dalam citra, yang dinyatakan oleh MN . M dan N merupakan ukuran baris dan kolom citra. Sedangkan normalisasi histogram pada persamaan $p(r_k) = n_k / MN$, $k = 0, 1, 2, 3, \dots, L-1$. Jadi $p(r_k)$ merupakan probabilitas jumlah sebuah level intensitas r_k dalam citra. Jika semua probabilitas r_k dijumlahkan maka hasilnya adalah 1.

3.6. Transformasi Wavelet Diskrit

Pada umumnya wavelet dimanfaatkan untuk mengeksplotasi kompresi citra. Tingkat kompresi yang didapat bergantung pada kekompakkan energi dari transformasi yang digunakan. Selain memiliki ciri kekompakkan energi yang tinggi dalam mengompresi citra, wavelet juga memiliki ciri rekonstruksi progresif yang menjadikannya tangguh dalam kompresi citra maupun video. Transformasi-transformasi yang ada kebanyakan memiliki fungsi-fungsi basis sinusoidal dan persegi panjang. Transformasi yang memiliki fungsi sinusoidal, seperti DFT, hanya dapat menganalisis atau mengidentifikasi frekuensi-frekuensi yang ada pada suatu

sinyal, tetapi tidak bisa mengidentifikasi waktu awal dan waktu akhir keberadaan frekuensi tertentu.

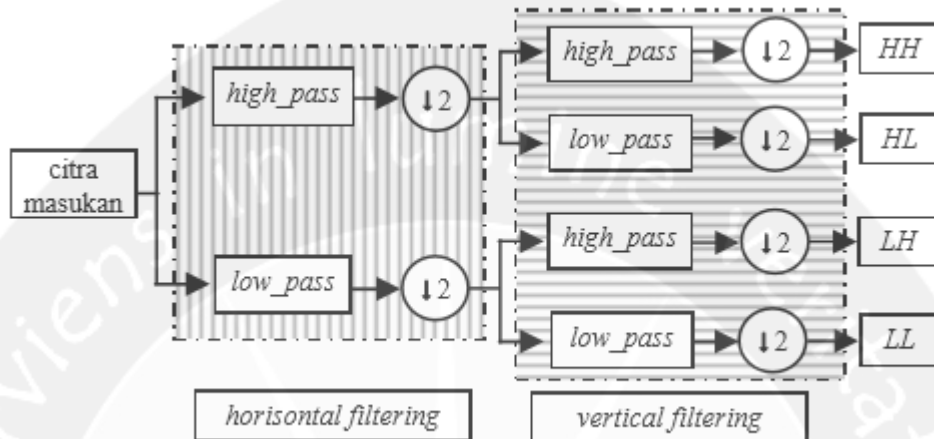
Transformasi dengan fungsi basis sinusoidal tidak bisa mengidentifikasi lokasi-lokasi dimana sesuatu terjadi. Fitur ini penting, dikarenakan ketika mengompresi suatu citra, area-area dengan aktivitas intens dan area-area datar dapat diidentifikasi sehingga alokasi bit kuantisasi untuk area-area tersebut dapat dibedakan yang pada akhirnya diperoleh kompresi yang tinggi tanpa mengorbankan kualitas visual.

Sama dengan transformasi fourier, transformasi wavelet juga memiliki dua versi, yaitu transformasi wavelet kontinyu (CWT, *continous wavelet transform*) dan transformasi wavelet diskrit (DWT, *discrete wavelet transform*). Dari segi komputasi, transformasi wavelet diskrit lebih tangguh dan hemat. Pada penelitian ini yang dibahas adalah DWT.

3.6.1. DWT 1-D

Pengkodean subpita pertama kali dibuat untuk memproses sinyal suara, cara ini berguna untuk mendekomposisi sinyal-sinyal suara ke dalam frekuensi pita oktaf. Selanjutnya pengkodean ini dikembangkan untuk mengompresi citra. Runtun 1D dianalisis memiliki panjang N untuk sementara. Pada tingkat pertama, runtun masukan difilter oleh dua filter, satu melalui filter *lowpass* dan yang satunya lagi melalui filter *highpass*. Keluaran filter menempati daerah frekuensi setengah bagian bawah dan setengahnya lagi bagian atas frekuensi pencuplikan. Ada kemungkinan terdapat dua keluaran filter yang saling tumpang tindih dikarenakan respon filter.

Maka pada tahap pertama terdapat dua keluaran dengan panjang masing-masing $N/2$, yang terkait dengan frekuensi rendah dan frekuensi tinggi.



Gambar 3. 4 Ilustrasi Transformasi Wavelet 1-D

Gambar diatas menampilkan langkah pertama yang dilakukan pada citra yang berukuran $m \times n$ adalah tahap tapis lolos-rendah yang merupakan *scaling function* dan tapis lolos-tinggi yang merupakan fungsi wavelet. Kemudian tahap selanjutnya turun pada dimensi m dengan faktor 2 yang akan memperoleh hasil citra lolos-rendah dan citra lolos-tinggi.

Selanjutnya masing-masing citra ditapis dan turun dengan faktor 2 sepanjang dimensi n . Akhir dari kedua proses ini akan dibagi kedalam sub bidang LL, HL, LH, HH. Dimana LL merupakan bidang perkiraan kasar atau aproksimasi dari citra asli, HL dan LH proses merekam perubahan pada citra sepanjang arah horizontal dan vertikal secara berurutan dan HH menunjukan p frekuensi tinggi yang terdapat pada citra.

3.6.2. DWT 2-D

DWT 2-D dapat diimplementasikan dengan operasi baris-kolom. Walaupun dapat diimplementasikan untuk mengimplementasi korelasi 2-D dalam suatu citra, namun akan lebih efisien untuk mengimplementasikan DWT 2-D menggunakan DWT 1-D, hal ini dikarenakan (a) banyak materi perancangan DWT 1-D tersedia di literatur; (b) implementasi cepat atas DWT 1-D juga tersedia.

3.7. Transformasi Wavelet Haar

Proses transformasi wavelet dapat dilakukan sampai level yang tak terhingga, tetapi dalam penerapannya sehari-hari proses transformasi wavelet dilakukan sampai proses detail data koefisien bernilai satu. Ini dikarenakan hasil informasi minimum memungkinkan untuk proses pengembalian sinyal asli, proses ini dinamakan *entropy*. Entropy berkorelasi dengan persamaan sebagai berikut:

$$Level_{maks} = \frac{\ln(\frac{panjang\ data}{panjang\ filter})}{\ln(2)}$$

Berdasarkan rumus diatas, untuk citra yang berukuran 128 x 128 piksel akan dialihragamkan menggunakan tapis wavelet berukuran 2 x 2, sehingga level maksimum yang diperbolehkan adalah 6 level. Wavelet haar merupakan wavelet yang paling sederhana dan paling tua diantara wavelet yang ada untuk menerapkan DWT 1-D dan DWT 2-D. Wavelet haar merupakan pencetus munculnya jenis wavelet lain seperti Daubechies dan lainnya.

Wavelet haar memiliki koefisien transformasi sebagai berikut :

$$h_0 = (h_0(0), h_0(1)) = (1/2, 1/2) \quad (\text{tapis lowpass})$$

$$h_1 = (h_1(0), h_1(1)) = (1/2, -1/2) \quad (\text{tapis highpass})$$

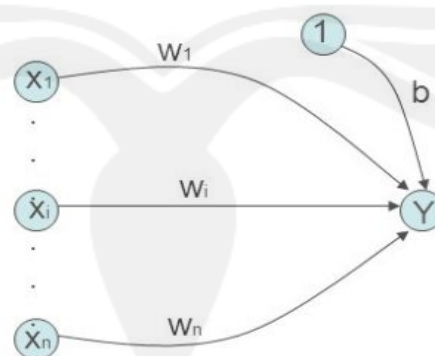
Kedua koefisien transformasi diatas bersifat orthogonal maupun tidak ortonormal. Kita dapat menggunakan tapis filter *lowpass* dan filter *highpass* untuk melakukan transformasi wavelet diskrit maju.

$$h_0 = \left(1/\sqrt{2}, 1/\sqrt{2}\right) \quad \text{koefisien filter lowpass}$$

$$h_1 = \left(1/\sqrt{2}, -1/\sqrt{2}\right) \quad \text{koefisien filter highpass}$$

3.8 Jaringan Syaraf Tiruan Backpropagation

Model jaringan perceptron ditemukan oleh Rosenblatt (1962) dan Minsky – Papert (1969). Model jaringan tersebut memiliki aplikasi dan pelatihan yang paling baik pada era tersebut. Perceptron memiliki arsitektur jaringan yang mirip dengan jaringan Hebb yaitu sebagai berikut :



Gambar 3. 5 Arsitektur Jaringan Perceptron

Jaringan perceptron terdiri dari beberapa unit masukan (ditambah sebuah bias), dan memiliki satu buah keluaran. Hanya saja fungsi aktivasi merupakan fungsi

biner (bipolar), tetapi masih memiliki kemungkinan untuk bernilai -1, 0, atau 1. Nilai threshold θ ditentukan oleh :

$$f(net) = \begin{cases} 1 & \text{jika } net > \theta \\ 0 & \text{jika } -\theta \leq net \leq \theta \\ -1 & \text{jika } net < -\theta \end{cases}$$

Secara geometris fungsi aktivasi membentuk 2 garis sekaligus, dimana masing-masing mempunyai persamaan :

$$W_1 X_1 + W_2 X_2 + \dots + W_n X_n + b = \theta, \text{ dan}$$

$$W_1 X_1 + W_2 X_2 + \dots + W_n X_n + b = -\theta$$

Dimisalkan S adalah vektor masuk nila, t adalah target keluaran, α adalah learning rate yang ditentukan, θ adalah threshold yang ditentukan, maka algoritma pelatihan perceptron adalah sebagai berikut :

- a. Inisialisasi semua bobot dan bias (umumnya $w_i = b = 0$), tentukan learning rate (α), supaya lebih sederhana nilai $\alpha = 1$
- b. Jika vektor masukan \neq target, maka lakukan :
 - 1) Set aktivasi masukan $x_i = s_i$ ($i = 1, \dots, n$)
 - 2) Hitung respon unit keluaran : $net = \sum x_i w_i + b$

$$f(net) = \begin{cases} 1 & \text{jika } net > \theta \\ 0 & \text{jika } -\theta \leq net \leq \theta \\ -1 & \text{jika } net < -\theta \end{cases}$$

- 3) Perbaiki bobot pola yang mengandung kesalahan ($y \neq t$) menurut persamaan :

$$W_i(\text{baru}) = w_i(\text{lama}) + \Delta w (i=1, \dots, n) \text{ dengan } \Delta w = \alpha t x_i$$

$$b(\text{baru}) = b(\text{lama}) + \Delta b \text{ dengan } \Delta b = \alpha t$$

Beberapa hal yang perlu diperhatikan dalam algoritma tersebut :

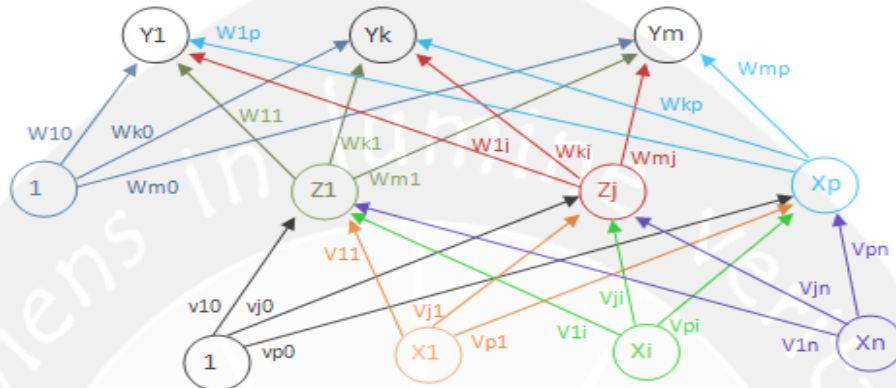
- a. Iterasi dilakukan terus hingga semua pola memiliki keluaran jaringan yang sama dengan target (jaringan sudah memahami pola). Iterasi tidak boleh berhenti setelah semua pola dimasukan seperti yang terjadi pada model Hebb.
- b. Perubahan bobot hanya dilakukan pada pola yang mengandung kesalahan (keluaran jaringan \neq target). Perubahan tersebut merupakan hasil kali unit masukan dengan *learning rate*. Perubahan bobot hanya akan terjadi jika unit masukan $\neq 0$.

3.8.1 Algoritma Backpropagation Umpan Balik (Feed Forward Backpropagation)

Propagasi umpan balik berbasis jaringan syaraf tiruan merupakan algoritma pembelajaran yang terawasi, biasanya digunakan perceptron dengan banyak lapisan untuk mengubah bobot-bobot yang terhubung dengan neuron-neuron yang ada pada lapisan tersembunyi. Propagasi umpan balik berbasis jaringan syaraf tiruan menggunakan *error output* untuk mengubah nilai bobot-bobotnya dalam arah mundur (*backward*). Untuk mendapatkan *error* ini, tahap *forward propagation* harus dikerjakan terlebih dahulu. Pada saat perambatan maju, neuron diaktifkan dengan menggunakan fungsi aktivasi yang dapat dideferensiasikan.

Feed forward merupakan algoritma berbasis JST yang memiliki beberapa unit yang ada dalam satu atau lebih *layer* tersembunyi. Gambar 3.7. merupakan gambar arsitektur *feed forward* yang memiliki n buah masukan (ditambah sebuah

bias), sebuah *layer* tersembunyi yang terdiri dari p unit (ditambah sebuah bias), serta m buah unit keluaran (Jek, 2005 : 98).



Gambar 3. 6 Arsitektur propagasi umpan balik berbasis JST

V_{ji} merupakan bobot garis dari unit masukan x_i ke unit layer tersembunyi z_i (v_{j0} merupakan bobot garis yang menghubungkan bias di unit masukan ke unit layer tersembunyi z_j). w_{kj} merupakan bobot dari unit layer tersembunyi z_j ke unit keluaran y_k (w_{k0} merupakan bobot dari bias di layer tersembunyi ke unit keluaran z_k) (Jek, 2005:98).

Pada proses pelatihan backpropagation ada beberapa parameter yang sangat berpengaruh, diantaranya adalah sebagai berikut :

1. Penentuan bobot awal , umumnya diambil secara acak dengan angka dalam jangkauan $[-0.5, +0.5]$ atau ditentukan dengan jangkauan $\left\{-\frac{2.4}{F_1}, +\frac{2.4}{F_1}\right\}$.
2. Laju pembelajaran (learning rate) jangkauan 0 sampai 1. Semakin besar nilainya semakin cepat selesai proses pelatihannya, namun semakin lebih mudah terjebak pada daerah local optima. Jika terlalu kecil proses pelatihan akan semakin lama, tetapi lebih menjamin hasil model yang lebih baik.

3. Kecilnya nilai laju pelatihan bisa mengakibatkan proses pelatihan yang lama, tetapi tidak menjamin hasil yang lebih baik, biasanya ditambahkan momentumnya untuk membantu mempercepat proses pencapaian target error tetapi dengan laju pelatihan yang kecil. Nilai momentum yang dipakai adalah antara 0 sampai 1.
4. Jika kriteria error hanya menggunakan SSE atau MSE, terkadang data yang sangat tidak linier sulit untuk mencapai kriterianya. Pilihan lain dalam kriteria biasanya menggunakan jumlah maksimal iterasi. Jika jumlah maksimal tercapai, meskipun target error belum tercapai, proses pelatihan tetap akan dihentikan.
5. Target error merupakan akumulasi selisih nilai antara nilai keluaran yang diharapkan dengan nilai keluaran yang didapatkan. Kriteria umum yang digunakan adalah *Sum of Square* (SSE) dan *Mean Square Error* (MSE).
6. Jumlah neuron dalam layer tersembunyi, untuk mendapatkan komposisi jumlah neuron dalam layer tersembunyi yang tepat biasanya digunakan dengan cara coba-coba sehingga dari beberapa kali percobaan akan diambil arsitektur yang memberikan hasil prediksi terbaik.

Karakteristik Backpropagation pada umumnya adalah :

1. Backpropagation merupakan ANN turunan perceptron yang menggunakan layer tersembunyi. Backpropagation dapat menangani masalah set data yang didistribusi kelasnya tidak linier dengan lebih baik dibandingkan perceptron, meskipun kadang-kadang hasil kinerja yang diberikan masih belum bisa maksimal.

2. Backpropagation mampu menangani fitur data yang redundan karena bobot yang identik akan dipelajari selama proses pelatihan. Bobot fitur yang redundan akan ditekan sehingga menjadi sangat kecil.
3. Backpropagation sensitif terhadap noise.
4. Proses pelatihan untuk pembentukan model dalam backpropagation tumbuh semakin banyak secara eksponensial setiap kali ada penambahan layer tersembunyi.
5. Dengan backpropagation, hasil keputusan model yang didapatkan bisa berbeda dengan setiap percobaan yang dilakukan. Hal ini disebabkan oleh prinsip kerja pelatihan dalam ANN, yaitu jika semua vektor masukan sudah berhasil dipetakan dengan benar, iterasi pelatihan dihentikan, tidak memperdulikan margin batas keputusan diantara dua kelas.

3.8.2 Momentum

Modifikasi metode penurunan tercepat dilakukan dengan menambah momentum. Perubahan dilakukan dengan merubah bobot yang didasarkan atas arah *gradient* pola terakhir dan pola sebelumnya (momentum) yang dimasukkan. Dengan demikian perhitungannya tidak hanya pola masukan terakhir saja.

Penambahan momentum dilakukan guna menghindari perubahan bobot yang mencolok akibat perbedaan data satu dengan yang lain. apabila data terakhir yang diberikan ke jaringan memiliki pola yang serupa (berarti arah *gradient* sudah benar), maka dilakukan perubahan bobot secara cepat. tetap jika data terakhir yang

dimasukkan memiliki pola yang berbeda dengan pola sebelumnya, maka perubahan bobot dilakukan secara lambat.

Proses penambahan momentum, merupakan proses bobot baru pada waktu $(t + 1)$ didasarkan atas bobot pada waktu t dan $(t - 1)$. Pada langkah ini ditambahkan dua variabel yang mencatat besarnya momentum untuk dua iterasi terakhir. Jika μ adalah konstanta $(0 \leq \mu \leq 1)$ yang menyatakan parameter momentum maka bobot baru dihitung berdasarkan persamaan berikut :

$$w_{kj}(t + 1) = w_{kj}(t) + \alpha \delta_k z_j + \mu (w_{kj}(t) - w_{kj}(t - 1))$$

$$v_{ji}(t + 1) = v_{ji}(t) + \alpha \delta_j x_i + \mu (v_{ji}(t) - v_{ji}(t - 1))$$